



Syntax and Semantics



Today's Lecture

- Compounds: Templates, Templates, ...
- Syntax
 - The acceptable form of what we write
 - Mathematically, a set $\langle \text{prog} \rangle$
 - Backus and Naur Form (BNF)
- Semantics
 - The meaning of syntax
 - Mathematically, well, we'll see



Shape = Triangle | Square

- *Shape* will combine two things:
 - (define-struct triangle (base height))
 - (define-struct square (side))
- Now we can talk about *shape* as either
 - (triangle b h) where b and h are numbers, or
 - (square s) where s is a number
- Functions that consume *shape* must follow a *template*



Template (for deconstruction)

; x : shape

```
(cond [(triangle? x)
      ... (triangle-base x)
      (triangle-height x)])
[(square? x)
 ... (square-side x)] )
```



Instance of template

; area : **shape** -> number

(define (area **x**)

 (**cond** [(**triangle?** x)

 (***** 0.5 (***** (**triangle-base** x)

 (**triangle-height** x))))]

 [(**square?** x)

 (**sqr** (**square-side** x)))])



Recall from lecture 3

- A syntactically correct program can be
 - An atom, like a number 17, `rabbit`, of a variable *radius*, or
 - A compound program,
 - starting with `(`
 - followed by operator (variable name)
 - then one (for now) or more programs
 - and, finally, ending with `)`
- Syntax errors: 3) , (3 + 4) , [+ 3 1],)+(...



Syntax

- For variables, constants, and primitives

<var> = x | area-of-disk | perimeter | ...

<con> = true | false | 'a | 'ringo ...

1 | -1 | 3/5 | 1.22 | ...

<prm> = + | - | ...



Syntax

- Now we can look at expressions:

$\langle \text{exp} \rangle = \langle \text{var} \rangle \mid \langle \text{con} \rangle \mid (\langle \text{prm} \rangle \langle \text{exp} \rangle \dots \langle \text{exp} \rangle)$
 $\mid (\langle \text{var} \rangle \langle \text{exp} \rangle \dots \langle \text{exp} \rangle)$
 $\mid (\text{cond } (\langle \text{exp} \rangle \langle \text{exp} \rangle) \dots (\langle \text{exp} \rangle \langle \text{exp} \rangle))$
 $\mid (\text{cond } (\langle \text{exp} \rangle \langle \text{exp} \rangle) \dots (\text{else } \langle \text{exp} \rangle))$

- And definitions:

$\langle \text{def} \rangle = (\text{define } (\langle \text{var} \rangle \langle \text{var} \rangle \dots \langle \text{var} \rangle) \langle \text{exp} \rangle)$
 $\mid ?$



Semantics

- Defined by explaining each syntactic entity in our language
 - `<exp>`
 - `<var>` means return the associated value
 - `<con>` means return this constant
 - `<def>`
 - (define ...) means introduce new variable, ...
- Read this chapter very carefully!